ML2020SPRING HW4 Report

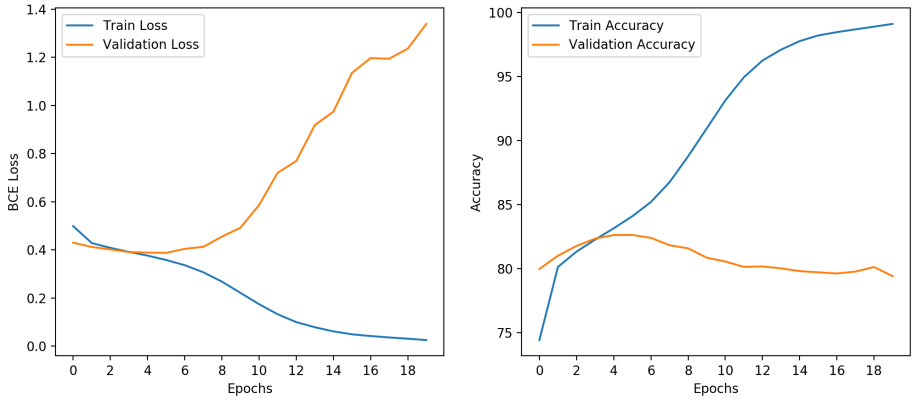
學號：R08946015 系級：資料科學碩一 姓名：陳鈞廷

1. 請說明你實作的 RNN 的模型架構、word embedding 方法、訓練過程 (learning curve) 和準確率為何?

我使用的 word embedding 方法是 skip-gram，使用 gensim 套件的 Word2Vec 實作，參數設定為： `size=100, window=5, min_count=5, workers=8, iter=10`。

我實作的 RNN 模型主要包含了兩層 `GRU`，設定 `hidden_dim` 為 512，並設定 `dropout` 0.4，最後接上一層 `linear` 作為 output。

下圖是我訓練了 20 個 epoch 的訓練曲線，我將 training example 的 0.15 比例作為 validation。可以發現第 5 個 epoch 開始出現了 overfitting 現象，因此我將第 6 個 epoch 的 model 拿來預測，得到的分數為 public: 0.82504 和 private: 0.82333。



2. 請比較 BOW + DNN 與 RNN 兩種不同 model 對於 "today is a good day, but it is hot" 與 "today is hot, but it is a good day" 這兩句的分數 (過 softmax 後的數值)，並討論造成差異的原因。

下方表格是分別用 RNN 和 BOW+DNN 預測這兩個句子的分數，可以 RNN 對這兩個句子的預測分數差距非常明顯，可能是因為 RNN 認為這兩個句子的 word 出現順序導致了不同的語氣；而另外可以看到 BOW + DNN 對於兩個句子的分數是一樣的，這是因為 BOW 只記憶句子中每個 word 出現的次數，並不包含 word 出現的順序。

	today is a good day but it is hot	today is hot but it is a good day
RNN	0.35269365	0.94189763
BOW + DNN	0.54655755	0.54655755

3. 請敘述你如何 improve performance(preprocess、embedding、架構等等)，並解釋為何這些做法可以使模型進步，並列出準確率與 improve 前的差異。(semi-supervised 的部分請在下題回答)

經過多次實驗，我發現 embedding 中的 vector size 對於準確率有較大的影響，當我將 vector size 縮減至 100 後，validation 的準確率上升了 0.01，我認為過大的 vector size 會讓 embedding space 過於 sparse，從而讓 embedding 的效果減小，使得模型辨識正確率下降。

	Validation Acc.	Public Score
Embedding Vector Size 300	0.818	0.81951
Embedding Vector Size 100	0.826	0.82504

另外我發現 GRU 的層數似乎對正確率的影響不太，因此為了避免過深的模型導致 overfitting，因此我將 GRU 的層數設定為 2。